Appendix B

Following is a pin definition file for a slave FPGA of a board according to an embodiment of the present invention.

```
/////////////////////////////////////////////////////////////////////////////
///
/// HEADER FILE FOR SLAVE FPGA - DEFINE FP1 IN THE MAIN SOURCE FILE
///
/////////////////////////////////////////////////////////////////////////////

#ifndef _KOMPRESSOR_SLAVE_HEADER
#define _KOMPRESSOR_SLAVE_HEADER

#warning Compiling design for the Slave FPGA

set part   = "XV2000e-6-FG680";
set family = Xilinx4000E;

/////////////////////////////////////////
// Clocks
/////////////////////////////////////////
// CLKA    D21
// CLKB    A20
```

```
// MCLK    AW19
// VCLK    AU22
// Only one clock is currently supported (HC2.1)

set clock = external_divide "D21" 2;

#define CLOCK_RATE 25000000  // 50MHz clock / 2

#define VGA // necessary for VGA driver

////////////////////////////////////////////
// Master Slave definition Pin
////////////////////////////////////////////

macro expr MS_define = { data = {"D33"}};

////////////////////////////////////////////////////////////////////////////
// Local SRAM definitions
////////////////////////////////////////////////////////////////////////////

/////////////////////////////
// Local SRAM BANK 0
//
// Though this bank is defined to be 32bits wide.
```

```
// it is possible to perform 8bit writes if required.
/////////////////////////////


macro expr DA_pins = {        "AA39", "AB35", "Y38", "AB36", "Y39", "AB37",
"AA36", "W39",
                                            "AA37", "W38",  "W37", "V39",  "W36",
"U39",  "V38",  "U38",
                                            "V37",  "T39",  "V36", "T38",  "V35",
"R39",  "U37",  "U36",
                                            "R38",  "U35",  "P39", "T37",  "P38",
"T36",  "N39",  "N38" };

macro expr AA_pins = { "R37", "M39", "R36", "M38", "P37", "L39", "P36", "N37",
                                            "L38", "N36", "K39", "M37", "K38",
"L37", "J39", "L36",
                                            "J38", "K37"};

macro expr CA_pins = {data = {"H39", "K36", "H38", "J37", "G39", "G38", "J36"}};



macro expr sram_local_bank0_spec =
 {
  offchip = 1,
      wegate = 1,
  data = DA_pins,
  addr = AA_pins,
  cs   = { "J36", "H38", "J37", "K36", "H39" },
```

```
we  = { "G38" },
        oe      = { "G39"}
};


////////////////////////////
// Local SRAM Bank 1
////////////////////////////

macro expr DB_pins = {     "AR37", "AR39", "AR36", "AT38", "AR38", "AP36",
"AT39", "AP37",
                                        "AP38", "AP39", "AN36", "AN38",
"AN37", "AN39", "AM36", "AM38",
                                        "AM37", "AL36", "AM39", "AL37",
"AL38", "AK36", "AL39", "AK37",
                                        "AK38", "AJ36", "AK39", "AJ37",
"AJ38", "AH37", "AJ39", "AH38"};

macro expr AB_pins = { { "AH39", "AG38", "AG36", "AG39", "AG37", "AF39",
"AF36", "AE38",
                                        "AF37", "AF38", "AE39", "AE36",
"AD38", "AE37", "AD39", "AD36",
                                        "AC38", "AC39"}};

macro expr CB_pins = {data = {"AD37", "AB38", "AC35", "AB39", "AC36", "AA38",
"AC37"}};

macro expr sram_local_bank1_spec =
```

```
{
 offchip = 1,
 wegate = 1,
      data = DB_pins,
 addr = AB_pins,
 cs   = { "AB38", "AD37", "AB39", "AC35", "AC37" },
 we   = { "AA38" },
      oe       = { "AC36"}
};
```

```
/////////////////////////////
// Shared SRAM definitions
/////////////////////////////
```

```
/////////////////////////////
// Shared SRAM BANK 0
//
// Though this bank is defined to be 32bits wide.
// it is possible to perform 8bit writes if required.
/////////////////////////////
```

```
macro expr SHAREDRAM0A_pins = {      "L1", "L2", "N3", "K1", "N4", "K2",
"M3", "J1",
```

```
                                              "L3", "J2", "L4", "H1",
"K3", "H2", "K4", "G1",
                                              "G2", "J3"};

macro expr SHAREDRAM0D_pins = {     "W1", "AB4", "AB3", "W2", "AB2",
"V1", "AA4", "V2",
                                              "AA3", "U1", "W3",  "U2",
"W4", "T1",  "V3",  "T2",
                                              "V4", "V5",  "U3",  "R2",
"U4", "P1",  "U5",  "P2",
                                              "T3", "N1",  "N2",  "T4",
"M1", "R3",  "M2",  "R4"};

macro expr sram_shared_bank0_request_pin = { data = { "A25" }};
macro expr sram_shared_bank0_grant_pin   = { data = { "B25" }};

macro expr sram_shared_bank0_spec =
 {
  offchip = 1,
  data = SHAREDRAM0D_pins,
  addr = SHAREDRAM0A_pins,
  cs   = { "E2", "H3", "F2", "J4", "F1"},
  we   = { "H4" },
       oe       = { "E1" }
 };
```

```
/////////////////////////
// Shared RAM bank1
/////////////////////////

macro expr SHAREDRAM1A_pins =        {"AG1", "AG4", "AF2", "AG3", "AF1",
"AF4", "AF3", "AE2",
                                      "AE4", "AE1", "AE3",
"AD2", "AD4", "AD1",  "AC1", "AB1",
                                      "AC5", "AA2"};

macro expr SHAREDRAM1D_pins = {       "AT3", "AP3", "AR3", "AT2", "AP4",
"AR2", "AT1", "AN4",
                                      "AR1", "AN3", "AP2",
"AN2", "AP1", "AM4", "AN1", "AM3",
                                      "AL4", "AM2", "AL3",
"AM1", "AL2", "AL1", "AK4", "AK2",
                                      "AK3", "AK1", "AJ4",
"AJ1", "AJ3", "AH2", "AJ2", "AH3"};

macro expr sram_shared_bank1_request_pin = { data = { "C25" }};
macro expr sram_shared_bank1_grant_pin  = { data = { "D25" }};


macro expr sram_shared_bank1_spec =
 {
  offchip = 1,
      wegate = 1,
  data = SHAREDRAM1D_pins,
```

```
addr = SHAREDRAM1A_pins,
cs   = { "AB5", "AC3", "Y1", "AA1", "AC4" },
we   = { "Y2" },
     oe        = { "AC2" }
};




/////////////////////////////////////////////////////////////
// ARM Interfacing Pins
/////////////////////////////////////////////////////////////


macro expr ARMA_pins = {data = { "C11", "B11", "C12", "A11", "D13",
                                                    "B12", "C13", "D14",
"A12", "C14"}};


macro expr ARMD_pins = {data = {"G3", "G4", "D2", "F3", "D3",
                                                    "F4", "D1", "C5", "A4",
"D6",
                                                    "B5", "C6", "A5", "D7",
"B6",
                                                    "C7", "A6", "D8", "B7",
"C8",
                                                    "A7", "D9", "B8", "A8",
"C9",
```

```
                                                  "B9", "D10","A9",
"B10","C10",
                                                  "D11", "A10"}};

macro expr ARMGPIO_pins = {data = {    "B34", "C33", "A34", "D32", "B33",
"C32",
                                                       "D31", "A33",
"C31", "B32", "B31"}};

macro expr ARMnWE_pin = { data ={"B13"}}; // input
macro expr ARMnOE_pin = { data ={"D15"}}; //input
macro expr ARMnCS4_pin = { data ={"A13"}}; // input
macro expr ARMnCS5_pin = { data ={"C15"}}; // input
macro expr ARMRDY_pin = { data ={"B14"}}; //ouput

////////////////////////////////////////////////////////////////////////////////
// Flash Memory interface  - may not be able to use definiton of Flash as a RAM if
// FPGA to FPGA configuration is required
////////////////////////////////////////////////////////////////////////////////

macro expr FA_pins = {      "E22", "B20", "D22", "C21", "B19", "C19", "A18",
"D19",
```

```
                                        "B18", "C18", "A17", "D18", "B17",
"E18", "A16", "C17",
                                        "D17", "B16", "E17", "A15", "C16",
"B15", "D16", "A14"};

macro expr FD_pins = {"AR4", "AH1", "AG2", "AD3", "R1", "P3", "P4", "C2"}; //
also to CPLD
macro expr FDH_pins = {"B24", "B22", "E23", "A22", "D23", "B21", "C23", "A21"};
// high byte of the RAM


macro expr FC_pins = {"D24", "A24", "B23", "C24", "A23"};//d // control pins | |oe|
|we|cs


macro expr flash_addr_spec =
 {
   offchip = 1,
   data = {},
   addr = FA_pins,
   cs   = { },
   we   = { },
         oe       = { }
 };

macro expr flash_data_spec =
 {
   offchip = 1,
   data = FD_pins,
```

```
  addr = {},
  cs  = { "A23"},
  we  = { "C25"},
      oe      = { "A24"}
};

macro expr flash_cs_pin = { data = {"A23"}};
macro expr flash_oe_pin = { data = {"A24"}};
macro expr flash_we_pin = { data = {"C25"}};

macro expr flash_sts_pin = {data = {"B23"}}; // status
macro expr flash_nByte_pin = {data = {"B24"}}; // x8 / x16 selector



////////////////////////////////////////////////
// Parallel Port interface
////////////////////////////////////////////////

macro expr PP_pins = {data = {      "G36", "D39", "D38", "F36", "D37",
                                    "E37", "C38", "B37",
"F37", "D35",
                                    "B36", "C35", "A36",
"D34", "B35",
                                    "C34", "A35"}}; // all the
pins
```

```
// ppo lines 12 11 10 9 8 6 4 2// pins 2 - 9 on the interface
macro expr pp_data_pins =  {data = { "D37", "E37", "C38", "B37",
                                                                 "F37", "B36",
"A36", "B35"}};


// Status Port - write to host
macro expr nAck_pin = { data = { "F36"}};       // ppo 13
macro expr busy_pin = { data = { "D38"}};       // ppo 14
macro expr pe_pin = { data = { "D39"}};         // ppo 15
macro expr select_pin = { data = { "G36"}};     // ppo 16
macro expr nError_pin = { data = { "D34"}};     // ppo 3

//busy @ nAck @ pe @ Select @ nError;
macro expr status_port_pins = { data = { "D38", "F36", "D39", "G36", "D34"}};

// Control Port  - read from host
macro expr nAutoFeed_pin = { data = { "C34"}};     // ppo 1
macro expr init_pin = { data = { "C35"}};          // ppo 5
macro expr nSelect_in_pin = { data = { "D35"}};    // ppo 7
macro expr nStrobe_pin = { data = { "A35"}};       // ppo 0

//nSelectin, init, nautofeed, strobe,
macro expr control_port_pins  = { data = { "D35", "C35", "C34", "A35"}};



/////////////////////////////////////////////
```

```
// LEDs - maybe declare subsets and allocate each FPGA some
// great care has to be taken if both FPGAs try to access the same LEDs
//////////////////////////////////////////////
macro expr LED_pins = {data = {     "AU13", "AT14", "AV12", "AU14",
                                    "AW12", "AT15", "AV13",
"AU15"}};



//////////////////////////////////////////////
// ATA Interface
//////////////////////////////////////////////
macro expr ATA_pins = {data = {     "AU26", "AV27", "AT26", "AW28", "AU27",
                                    "AV28", "AW29", "AT27",
"AW30", "AU28",
                                    "AV30", "AV29", "AW31",
"AU29", "AV31",
                                    "AT29", "AW32", "AU30",
"AW33", "AT30",
                                    "AV33", "AU31", "AT31",
"AW34", "AV32",
                                    "AV34", "AU32", "AW35",
"AT32", "AV35",
                                    "AU33", "AW36",
"AT33"}};
```

```
/////////////////////////////////////////////////////
// Expansion Bus (32 bits)
///////////////////////////////////////////////////////

macro expr E_pins = {data = {        "AV17", "AU18", "AW17", "AT19", "AV18",
                                     "AU19", "AW18", "AU21",
"AV19", "AW20",
                                     "AV20", "AR22", "AV23",
"AW21", "AU23",
                                     "AV21", "AT23", "AW22",
"AR23", "AV22",
                                     "AV24", "AW23",
"AW24", "AU24", "AW25",
                                     "AT24", "AV25", "AU25",
"AW26", "AT25",
                                     "AV26", "AW27"}};


///////////////////////////
// Serial H Bus
///////////////////////////
macro expr SERIALH_pins = {data = {"F39", "H37", "F38", "H36", "E39", "G37",
"E38"}};


////////////////////////////////////////
```

```
// SelectLink Bus - Directly connects the 2 FPGAs
////////////////////////////////////////

macro expr SL_pins = {data = {     "AV3", "AU4", "AV5",  "AT6", "AV4",  "AU6",
                                                "AW4", "AT7", "AW5",
"AU7", "AV6",  "AT8",
                                                "AW6", "AU8", "AV7",
"AT9", "AW7",  "AV8",
                                                "AU9", "AW8", "AT10",
"AV9", "AU10", "AW9",
                                                "AT11","AV10","AU11",
"AW10","AU12", "AV11",
                                                "AT13", "AW11"}};

//////////////////////////
//VGA interface
//////////////////////////
macro expr VGA_pins = {data = {     "AW13", "AV14", "AT16", "AW14", "AU16",
                                                "AV15", "AR17", "AW15",
"AT17", "AU17",
                                                "AV16", "AR18", "AW16",
"AT18"}};

macro expr vga_vsync_pin = { data = { "AV14" } };
macro expr vga_hsync_pin = { data = { "AW13" } };
macro expr vga_data_pins = { data = {  "AT16", "AW14", "AU16", "AV15",
```

```
                    "AR17", "AW15", "AT17", "AU17",
                    "AV16", "AR18", "AW16", "AT18"} };

// macros for compatibility with existing programs
macro expr vsync_pin = { "AV14"  };
macro expr hsync_pin = { "AW13" };
macro expr video_spec = { data = {   "AT16", "AW14", "AU16", "AV15",
                    "AR17", "AW15", "AT17", "AU17",
                    "AV16", "AR18", "AW16", "AT18"} };

//////////////////////////
// CPLD interface pins
//////////////////////////

macro expr BUSMaster_pin = { data = { "C26" }}; // P12
macro expr FPcom_pins = { data = { "B26", "C27", "A27"}}; //P14 P15 P16

//////////////////////////
// Serial Ports pins
//////////////////////////

macro expr SERIAL_pins = {data = {"AV36", "AU34", "AU36", "AT34"}};

macro expr rs232_txd_pin = {data = { "AV36"}};
macro expr rs232_rxd_pin = {data = { "AU36"}};
```

```
macro expr rs232_rts_pin = {data = { "AU34"}};
macro expr rs232_cts_pin = {data = { "AT34"}};

////////////////////////////////////////////////////////////////////
// USB
////////////////////////////////////////////////////////////////////

macro expr USBMaster_pin = { data = { "D26" }}; // P13

macro expr USBD_pins = {data = {"C29", "A30", "D29", "B30", "C30", "A31", "D30",
"A32"}};

macro expr USBMS_pins = { data = {"D27"} };

macro expr USBnRST_pins = { data = {"B27"} };

macro expr USBIRQ_pins = { data = {"C28"} };

macro expr USBA0_pins = { data = {"A28"} };

macro expr USBnRD_pins = { data = {"B28"} };

macro expr USBnWR_pins = { data = {"B29"} };

macro expr USBnCS_pins = { data = {"A29"} };
```

```
#endif _KOMPRESSOR_SLAVE_HEADER
```